[0010] A digital signature may be generated in other ways as well. For example, instead of digitally signing the data, the signatory can digitally sign a hash or digest of the data. A hash or digest is obtained by operating a hash algorithm on the data file. A hash algorithm is a method of transforming a variable length message, in this case the data file, into a fixed length number. This fixed length number is referred to as the hash or digest of the original data file. For this digest to be useful as part of a digital signature, the contents of the data file must not be practically ascertainable from the digest number. Thus, hash algorithms are one-way functions, which can easily generate a hash from a data file, but which cannot, for all practical purposes, generate the original data file given the hash. The digest's usefulness as a digital fingerprint of a data file also depends upon its ability to correlate uniquely to the original data file. Ideally, a hash algorithm is a strictly one-to-one function so that each hash number can be generated by one, and only one, data file. Any change in the data file, no matter how insignificant, will generate a different hash number. If a hash algorithm generates the same hash for two different data files, a collision exists which could compromise the usefulness of the hash. Thus, one measure of a hash algorithm's usefulness is the frequency at which more than one data file will generate the same hash number. In practice, useful hash algorithms may generate collisions in theory but the probability is low enough as to be practically negligible. Well-known one-way hash algorithms that are useful for digital signing include MD2, MD5, and SHA-1.

[0011] The hash of the data file, along with information about the hash algorithm used to generate the hash, is then encrypted with the signatory's private key. The signatory provides the original data file as well as the encrypted hash to the recipient. The recipient uses the signatory's public key to decrypt the hash. To verify the integrity of data, the recipient uses the same hash algorithm on the original data file. If the hash generated by the recipient does not match the decrypted hash, this indicates a problem. The digital signature may not have been created with the signatory's private key or the data may have been tampered with since the signatory signed it. If the hashes match, the recipient can be reasonably assured that the signatory signed the data

and that it has not been altered. For the following discussion of the present invention, references to digital signatures or digitally signing shall include all of the aforementioned variants of the digital signatures and digitally signing.

[0012] Although the technology exists to create digital signatures, there are several challenges for a practical digital signature system. For example, because personal and business users work with various applications and with various types of documents, each of which may require a signature or signatures, a universal solution requires support for digital signing of any digital data, regardless of the file format. Also, many transactions, particularly business transactions, require support for multiple signatures and easy exchange of files and digital signatures. Furthermore, users require effective archiving that binds a digital signature or digital signatures with the signed digital data.

[0013] Current technology allows a digital signature to be generated for digital files, but there does not exist a universal object that will bind digital signatures to digital data, regardless of the file format. For example, word processor plug-ins are available which allow documents in Microsoft Word format to be digitally signed, but such functionality is not available for all applications and file formats. In addition, other digital signature services store signatures online but do not bind them to the original content for archiving. Nor do these services easily support countersigning.

[0014] What is needed is a universal signature object that can bind digital signatures to digital data, regardless of the file format. With such an object, people and businesses could more easily exchange documents and countersign data, such as contracts, without reverting to hard copies. Furthermore, with such an object, the digital data and all digital signatures can easily be archived.

SUMMARY OF THE INVENTION

[0015] In accordance with the present invention, there is provided a universal signature object (100) for binding digital data (200) to at least one digital signature (112). In an embodiment, the universal signature object (100) contains a version (102, 103, or 104) of the digital data (200), information (106) concerning an application compatible with a file format of at least one of the versions (102, 103, 104), and signature information (108) of at least one signatory. The signature information (108) of a signatory contains at least one digital signature (112) of signature data (570), which is functionally related to the digital data (200).

[0016] In one variant, the signatory information (110) also contains timestamp information (116). In another embodiment, the signature information (110) contains information about the signatory's public key (118). In yet another embodiment, the universal signature object (100) includes use-permission information (130) indicating how a version or versions of the digital data (200) can be utilized. Alternatively, the universal signature object (100) includes a universal-signature-object viewer (600) for utilizing the universal signature object (100) to generate and display information from or related to the universal signature object (100). In an embodiment, the universal signature object (100) includes a signing program (400), which is an executable file used to generate a universal signature object (100) or to append a digital signature to an existing universal signature object (100).

[0017] In another aspect of the invention, a universal-signature-object viewer (600) includes an application launching means (602) and a viewer means (604). The application launching means (602) launches an application compatible with a file format of a version of the digital data (200). The viewer means (604) generates information concerning the universal signature object (100) for display to a user of a USO viewer (600). In an embodiment, the USO viewer (600) also contains an edit disabling means (606) for disabling the edit capabilities inherent in an application launched by the application launching means (602). In another embodiment, a verification means (608) verifies one or more of the digital signatures included in